**Journal of Computer and Communications, 2014, 2, 112-120**
Published Online July 2014 in SciRes. http://www.scirp.org/journal/jcc
http://dx.doi.org/10.4236/jcc.2014.29015

# Speech Signal Recovery Based on Source Separation and Noise Suppression

**Zhe Wang, Haijian Zhang, Guoan Bi**

School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
Email: wang0755@e.ntu.edu.sg, zhaijian@ntu.edu.sg, egbi@ntu.edu.sg

Received May 2014

## Abstract

**In this paper, a speech signal recovery algorithm is presented for a personalized voice command automatic recognition system in vehicle and restaurant environments. This novel algorithm is able to separate a mixed speech source from multiple speakers, detect presence/absence of speakers by tracking the higher magnitude portion of speech power spectrum and adaptively suppress noises. An automatic speech recognition (ASR) process to deal with the multi-speaker task is designed and implemented. Evaluation tests have been carried out by using the speech database NOIZEUS and the experimental results show that the proposed algorithm achieves impressive performance improvements.**

## Keywords

**Speech Recovery, Time-Frequency Source Separation, Adaptive Noise Suppression, Automatic Speech Recognition**

## 1. Introduction

In ubiquitous environment of multiple speakers, it has been a challenge to adapt the speech recognition model correctly for improving the speech recognition accuracy. The recovery of clean speech from a noisy resource is of vital importance for speech enhancement, speech recognition and many other speech related applications. In real life, there are numerous noise sources such as environment, channel distortion and speaker variability. Therefore, many algorithms have been reported for removing the noise from speech. Most of these algorithms need additional noise estimation and are only adapted to auditory effect rather than the ASR. This paper describes a novel self-optimized voice activity detection (VAD) algorithm together with a simple but effective noise removing process after signal separation for improving speech recognition rate. The key feature of the proposed VAD algorithm is that prior estimation of clean speech variance is not needed. In addition, the threshold used for speech/non-speech decision is generated from the noisy speech itself, which is considered as a kind of self-optimizing process. For the noise removing process, the key feature is the simplicity because it is based on the widely known spectral subtraction (SS) [1] method without any additional model or training process. Performance comparison has been made among SS method [2], zero-crossing-energy method (ZCE) [3], entropy-based method [4], and the proposed VAD based algorithm.

**How to cite this paper:** Wang, Z., Zhang, H.J. and Bi, G.A. (2014) Speech Signal Recovery Based on Source Separation and Noise Suppression. *Journal of Computer and Communications*, **2**, 112-120. http://dx.doi.org/10.4236/jcc.2014.29015

To perform the recognition task simultaneously, a modified recognition speaker process based on Bhattacharyya distance is proposed to process the separated speech for isolate word recognition. In this recognition scenario, the computational complexity is not increased in proportion to the number of template words from multiple speakers. Experimental results based on the noise and speech from the NOIZEUS database show the desirable performance.

The rest of the paper is organized as follows. In Section 2, a speech separation process is described to obtain individual signals from the noisy voice of multiple speakers without knowing the number of speakers. In Section 3, the self-optimized VAD algorithm is presented with detailed derivation steps. Section 4 provides an adaptive soft decision algorithm for noise suppression. Section 5 presents the modified Mel Frequency Cepstrum Coefficient (MFCC) based ASR system used in the multi-speaker adaption experiment. The experimental results and the performance comparison among various reported methods are presented in Section 6. Conclusions are given in Section 7.

## 2. Blind Source Separation (BSS) Based on SSTFT

Assume $S_n(t)$, n = 1, …, N, be the unknown speech sources, where N is the number of speakers. The arrangement of an M-sensor microphone array is linear. The output vector $x_m(t), m = 1, ..., M$ is modeled as

$$\mathbf{x}(t) = \mathbf{A}\,s(t) + \mathbf{n}(t) \tag{1}$$

where **A** denotes the mixing matrix, $\mathbf{x}(t) = [x_1(t), \ldots, x_M(t)]^T$ is the vector of the received mixtures, $\mathbf{s}(t) = [s_1(t), \ldots, s_N(t)]^T$ contains the multiple speech sources, n(t) is the additive white noise vector and T is the transpose operator.

The procedure of the spatial short-time Fourier transform (SSTFT) BSS algorithm based on the above signal model is presented as follows:

- Calculating the STFT of the mixtures **x**(t) in (1), we obtain an M × 1 vector $\mathbf{S}_x(t, f)$ at each time-frequency (TF) point $(t, f)$ as

$$\mathbf{S}_\mathbf{x}(t, f) = \mathbf{A}\mathbf{S}_\mathbf{s}(t, f) + \mathbf{S}_\mathbf{n}(t, f) \tag{2}$$

  where the subscript, S, denotes the STFT operator.
- Next, we detect the auto-source TF points, i.e. the auto-term location of the speech sources in TF domain based on the criterion at each time-instant

$$\| S_x(t, f) \| / \max_v \| S_x(t, v) \| > \varepsilon_0 \tag{3}$$

  where ||•|| denotes the norm operator and $\varepsilon_0$ is an empirical threshold value for selection of the auto-source TF points. All the TF points which satisfy the criterion in (3) are included in the set Ω.
- The premise of the SSTFT-based algorithm is to estimate the number of sources N as well as the mixing matrix A. We apply the method proposed in our previous work [5]. Specifically, we try to detect some dominant TF points, *i.e.*, the points at which one of the sources has the dominant energy compared to those of other sources and noise power. The mean-shift clustering method [6] is applied to classify the dominant TF points without knowing the number of sources. The mixing matrix **A** is estimated by averaging the spatial vectors of all TF points in the same cluster, and N is estimated by counting the number of resultant clusters.
- Based on the detected auto-source TF point set Ω and the estimated mixing matrix **A**, we can apply the subspace-based method to estimate the STFT values of each source [7]. We assume that there are at most K < M sources present at each auto-source TF point $\in$ Ω. Thus, the expression in (2) is simplified as

$$\mathbf{S}_\mathbf{x}(t, f) \approx \tilde{\mathbf{A}}\tilde{\mathbf{S}}_\mathbf{S}(t, f), \quad (t, f) \in \Omega \tag{4}$$

- where $\tilde{\mathbf{A}}$ denotes the steering vectors of the K sources at each point $(t, f) \in \Omega$, and $\tilde{\mathbf{S}}_\mathbf{S}(t, f)$ contains the STFT values of these K sources. $\tilde{\mathbf{A}}$ at each auto-source TF point can be determined by the following minimization

$$\tilde{\mathbf{A}} = \arg_{\hat{a}_{m_1}, \ldots, \hat{a}_{m_K},} \{PS_X(t, f)\} \tag{5}$$

  where $\mathbf{P} = \mathbf{I} - \tilde{\mathbf{A}}_m(\tilde{A}_m^H \tilde{A}_m)^{-1} \tilde{A}_m^H$ denotes the orthogonal projection matrix into noise subspace, where $\tilde{A}_m = [\hat{a}_{m_1}, \ldots, \hat{a}_{m_K}]$ contains the random K columns of the estimated mixing matrix **A**. The K STFT values at

each auto-source TF point can be obtained by

$$\tilde{\mathbf{S}}_s(t,f) \approx \tilde{\mathrm{A}}^{\dagger}\mathbf{S}_x(t,f),\ (t,f) \in \Omega \tag{6}$$

where † denotes the Moore-Penrose's pseudo inversion operator. Now the energy at each auto-source TF point in Ω is separated into K STFT values assigned to corresponding sources.

- Each source is recovered by the inverse STFT [8] using the estimated STFT values by (6).

**Figure 1** presents the results of using the above presented process for separating four speakers' voices from the signals received by two microphones, as seen in **Figure 2**.

## 3. Noise Estimation

Noise and speech are usually statistically independent and possess different statistical properties. Noise is more symmetrically distributed and present all the time, while speech is frequently non-stationary due to its active/non-active periods. The active/non-active transition of speech makes the energy of speech more concentrated in the speech active period.

### 3.1. General Description

The different behaviors of noise and speech make it possible to track speech or noise based on the minimum/

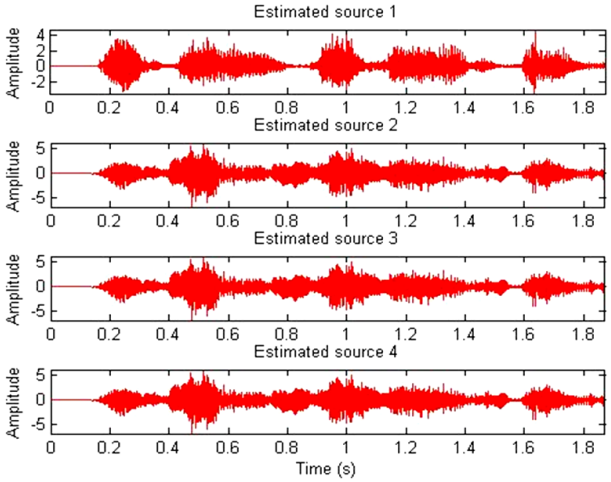


**Figure 1.** Separated voices from four speakers.

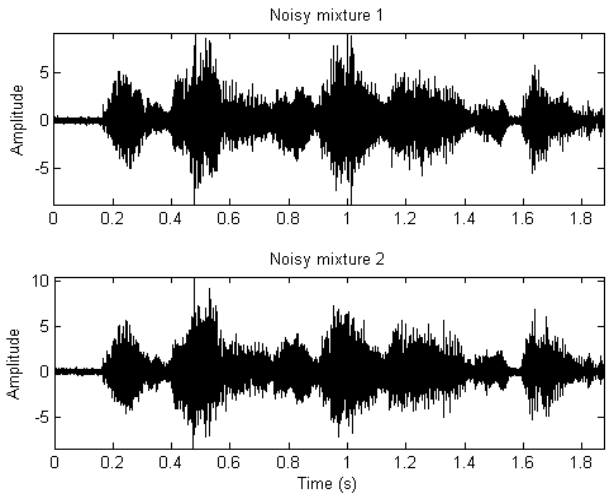


**Figure 2.** Mixed speech signals received by two microphones.

maximum of speech spectrum. The part having high energy is more likely to be speech while the part with low energy is more likely to be noise, which makes it possible to detect speech by analyzing the maximum of noisy speech. The speech amplitude is more probably larger than the noise amplitude. Compared with noise, the probability distribution function of clean speech magnitude is flatter in the "tail" part, which means clean speech magnitude is more likely far from its average. Even for SNR = 0 dB, the peak portion of the signal can be proven to be more likely from speech.

### 3.2. Algorithm Derivation

Assuming that speech is distorted by uncorrelated additive noise, two hypotheses for VAD are

$H_0$ : speech absent: Y = N + R

$H_1$ : speech present: Y = S + N + R

where Y, N, S, and R denote the frequency domain noisy speech, noise, clean speech and residual speech from the source separation process, respectively. The probability density function for $H_0$ and $H_1$ are given by

$$H_0 : P_N(Y) = f\left(Y \middle| H_0\right) = \frac{exp\left(-\frac{|Y|^2}{2\sigma_N^2}\right)}{2\pi\left(\sigma_N^2 + \sigma_R^2\right)} \tag{7}$$

$$H_1 : P(Y) = f\left(Y \middle| H_1\right) = \frac{exp\left[-\frac{|Y|^2}{2\left(\sigma_N^2 + \sigma_R^2 + \sigma_S^2\right)}\right]}{2\pi\left(\sigma_N^2 + \sigma_R^2 + \sigma_S^2\right)} \tag{8}$$

where $\sigma_N^2$, $\sigma_R^2$ and $\sigma_S^2$ denote the noise, residual and clean speech variance.

We have the conditions $\mathrm{P_N}(\mathrm{Y})/\mathrm{P}(\mathrm{Y}) < \varepsilon$ and $\mathrm{P_S} \gg \mathrm{P_N}$, where $\varepsilon$ is a heuristic parameter between 0.01 and 0.2. Its calculation will be further presented in Section 4. The first condition is simplified into

$$\sqrt{\frac{\left(\sigma_S^2 + \sigma_N^2 + \sigma_R^2\right)}{\sigma_N^2 + \sigma_R^2}} \exp\left[-\frac{|Y|^2}{2}\left(\frac{1}{\sigma_N^2 + \sigma_R^2}\frac{1}{\sigma_N^2 + \sigma_R^2 + \sigma_S^2}\right)\right] < \varepsilon \tag{9}$$

Define $\sigma_S^2 / \sigma_N^2 + \sigma_R^2 = k$, then

$$\sqrt{k+1}\exp\left[-\frac{|Y|^2}{2\left(\sigma_N^2 + \sigma_R^2\right)(1+k)}(k+2)\right] < \varepsilon \tag{10}$$

or equivalently,

$$\exp\left[-\frac{|Y|^2}{2\left(\sigma_N^2 + \sigma_R^2\right)(1+k)}\right] < \left(\varepsilon\sqrt{\frac{1}{k+1}}\right)^{\left(\frac{1}{(k+2)}\right)} \tag{11}$$

From (11), we have

$$|Y|^2 > -\frac{2\left(\sigma_S^2 + \sigma_N^2 + \sigma_R^2\right)}{k+2}\ln\left(\varepsilon\sqrt{\frac{1}{1+k}}\right) \tag{12}$$

Then define

$$|Y_\varepsilon|^2 = -\frac{2\left(\sigma_S^2 + \sigma_N^2 + \sigma_R^2\right)}{k+2}\ln\left(\varepsilon\sqrt{\frac{1}{1+k}}\right) \tag{13}$$

where $Y_\varepsilon$ can be served as a more direct threshold. Then the frequency bin level VAD flag can be achieved by

$$\text{flag} = \begin{cases} 1, & |Y|^2 > |Y_\varepsilon|^2 \\ 0, & |Y|^2 \le |Y_\varepsilon|^2 \end{cases} \tag{14}$$

The speech probability density function is calculated, which can be used to get $|Y_\varepsilon|^2$. Then, the binary VAD flag can be achieved by using (13). The VAD algorithm mentioned previously is suitable on suppressing the noise and can effectively distinct the noise and voiced speech. To improve the auto speech recognition rate, we still need to trace changes in the noise-energy and update the noise energy in all frames including the speech frames.

## 4. Noise Suppression

In the design of VAD algorithm, the soft decision algorithms appear to be superior to binary decision because speech signal is highly non-stationary. There is no clear boundary which marks the beginning or ending of a pronunciation. In this section, the discrimination information is used as a soft decision threshold.

### 4.1. Sub-Band Energy Calculation

The energy calculation works on a frame by frame basis. Each frame is multiplied by a suitable window to reduce the frequency aliasing from fast Fourier Transform (FFT). Note that the 50% overlapping means an initial delay of one half of the frame size is incurred. The frame size should be selected carefully. Suppose the sample rate is $F_S$ and frame size is $N = 2^m$. The time resolution is $N/F_s$, and frequency resolution is $F_s/N$. Obviously a larger frame size allows better frequency resolution but has a poor time resolution. In general, for $F_s = 8000$ and 16,000 Hz, frame sizes of N = 256 and 512 are found appropriate, respectively.

The signal is divided into 16 sub-bands. When the frame size is 256, the energy for the *i*th band is

$$S_i = \sum_{k=16i-15}^{16i} R_{i,k}^2 \tag{15}$$

where $R_{i,k}$ is the absolute value of the $k^{th}$ Fourier transform coefficient of the $i^{th}$ band. The sub-band out of the whole energy is calculated by

$$P_i = S_i \Big/ \sum_{j=1}^{16} S_j \tag{16}$$

The frame energy and the sub-band energy are used to calculate the discrimination information based on the sub-band energy distribution probabilities for both the current frame and the noise frame.

Assume the random variable Y has the chance to be a value of $a_1, a_2, ..., a_k$. The probability distribution of Y is related to the hypothesis $H_0$ and $H_1$. Set $P_0(\alpha_k) = P(\alpha_k|H_0)$, $P_1(\alpha_k) = P(\alpha_k|H_1)$, the discrimination information is defined as follows,

$$I(P_1, P_0; Y) = \sum_{k=1}^{K} P_1(a_k) \log[P_1(a_k) / P_0(a_k)] \tag{17}$$

The discrimination information can be calculated using sub-band energy distribution to measure the similarity of current frame and the noise frame.

$$P_0(a_k) = N_k / \sum_{j=1}^{8} N_i \tag{18}$$

$$P_1(a_k) = S_k / \sum_{j=1}^{8} S_i \tag{19}$$

### 4.2. Threshold Update

The threshold value is updated by:

- The first 5 frames are selected as the noise/non-speech frames.
- The previous frame of a period of speech signal is considered as noise frame.
- When the previous frame is considered as a noise frame, the current frame will be considered as noise frame if current frame satisfies $|Y|^2 \le |Y_\varepsilon|^2$. If the current frame satisfies $|Y|^2 > |Y_\varepsilon|^2$ and $d > T_r$, the current frame is considered as the start position frame and comparing with the next 6 frames is made. If the 6 frames also satisfy $|Y|^2 > |Y_\varepsilon|^2$ and $d > T_r$, the start position frame can be taken as the start position of a period of speech. Otherwise the current frame is still considered as a noise frame.
- When the previous frame is a speech frame, if current frame satisfies $|Y|^2 > |Y_\varepsilon|^2$, it remains to be the speech

frame. If the current frame satisfies $|Y|^2 \le |Y_\varepsilon|^2$ and $d < T_r$, it is classified as the end position frame, and then comparison with the next 6 frames is made. If the 6 frames also satisfy $|Y|^2 \le |Y_\varepsilon|^2$ and $d < T_r$, the end position frame can be taken as the end position of a period of speech (also the start point of a speech); otherwise, the current frame is still a speech frame. $T_r$ is the edge value of the discrimination information which equals to the average discrimination value of the most recently 5 noisy frames.

- During each step of the above determination, the noise threshold will be updated by

$$TH_n = TH_{n-1}(1-\lambda) + |Y|^2 \lambda \tag{20}$$

where $TH_n$ represents the updated noise threshold for the nth frame, $|Y|^2$ is the probability distribution function value of current speech and $\lambda$ is the noise update factor, which is calculated by discrimination information.

- If all the data have been dealt with, the adaptive adjustment will end.

### 4.3. Modified VAD and Noise Suppression

The speech signal $Y(w)$ is generally corrupted by the additive noise, $N(w)$, which is assumed to be independent of speech. In theory, the noise can be optimally removed by estimating its power and filtering the noisy signal with the following filter:

$$H(w) = \left(|Y(w)| - |N(w)|\right) / Y(w) \tag{21}$$

The proposed VAD will detect the noise frame, and subtract noise spectra from speech signal, trying to keep more information used during the feature extraction process of ASR and eliminating noise that provides wrong information during feature extraction and template matching. As the speech signal is always non-stationary, making a binary decision of being voice or noise is quite risky. Therefore, we have designed a module that rates the likelihood of voice by computing a voice activity score (VAS). In this way, we can achieve smooth processing transition when the derived VAS indicates a mixture of voice and noise.

The VAS for a frame is determined by two aspects. The first one concerns with the intelligibility of the voice, which is approximately quantified by counting the number of Bark bands in the speech band whose power, exceeds that of the corresponding Bark band of the estimated noise. The speech band ranges from the 4th to the 14th Bark band. The second aspect is the relative power of the current frame to that of the estimated noise power. In general, the higher the relative power of a frame, the more likely it contains voice. The final VAS is simply the sum of the scores from the two aspects. The parameter $\varepsilon$ is set as the reciprocal of VAS and is updated for each frame. The continuous VAS offers much more flexibility than a fixed parameter. Even when it is necessary to make a binary decision as to whether or not the frame is a noise-only frame, we can still make the processing changing and converging at certain value.

Then for each frame, the process described in Section 3.2 is conducted. With the processed results, $\sigma_N$ and $\sigma_S$ mentioned in Section 3.2 are updated correspondingly.

## 5. Speaker and Speech Recognition

This section elaborates the full system from the front end feature extraction, training process which consists of sub-word unit and word template generation, and the final recognition process. After VAD and noise suppression, the processed speech signal will be evaluated in this ASR system.

### 5.1. Front end Feature Extraction

The feature vector used for this recognition task is 24MFCC. The frame window size is 20 ms and the speech is sampled at 16 kHz with 16 bit resolution.

### 5.2. Sub-Word Unit Generation

The first part of the training process requires the users to record approximately two minutes of their speech. It is recommended to read phonetically rich sentences in order to obtain a more comprehensive sub-word unit. In this experiment, the user is asked to read a series of Harvard sentences. Then, the resulting MFCC is clustered by using c-means algorithm into 64 distinct units, roughly corresponding to a collection of sub-word. Each of these

clusters is then modeled using Gaussian mixture model of 4 mixtures. In this experiment, re-clustering is not done during word template generation. To simplify the model, further calculation is performed to generate the 64 by 64 Bhattacharyya distance matrix. This process is illustrated in **Figure 3**.

### 5.3. Word Template Generation

In this step, the words to be recognized are registered. As shown in **Figure 4**, the user is asked to pronounce the words, and the template generation converts those words into a sequence of sub-word unit index (obtained from the previous step) based on its maximum likelihood. To avoid over segmentation of the word, transitional heuristic is employed by allowing the change of sub-word index only when there is a significant margin of likelihood difference with the neighboring state. This process has to be repeated for each word that the user wants to introduce to the system.

### 5.4. Matching Process—Recognition

Assuming there are M word templates in the system, the recognition process calculates the probability of the user input feature vector X input being generated by the template. The chosen word is the one which gives the highest likelihood.

$$m^* = \arg\max\ p_m(\mathrm{X}\,input) \tag{22}$$

Note that the template can be viewed as a sequence of Gaussian Mixture Models (GMM), which makes the $p_m(X_{Input})$ calculation increasingly expensive with an increasing number of word templates and very hard to observe the effect of proposed VAD algorithm [9]. We propose to convert the input feature to a sub-word unit index sequence using the Bhattacharyya distance. The Bhattacharyya between two probability distribution $p_1$ and $p_2$ is defined by

$$\mathrm{B}batt\left(p_1 \middle| p_2\right) = -\ln\left(\int \sqrt{p_1(x)}\sqrt{p_2(x)}dx\right) \tag{23}$$

Each sub-word unit in the testing experiment is modeled by using 4 mixtures GMM, so the distance between them is given by:

$$\mathrm{B}batt\left(p_1 \middle| p_2\right) = -\ln\left(\int \sqrt{\sum_{mix=0}^{3} p_{1,mix}(x)}\sqrt{\sum_{mix=0}^{3} p_{2,mix}(x)}dx\right)$$

The distance is calculated for all 64 sub-word units using Levenshtein distance method. The average run time of the recognition task by original pattern matching algorithms increases proportionally with the number of templates. For the Bhattacharyya edit distance method, however, the running time is quite stable when the number of templates increases, which is particularly suitable for a real-time recognition system. **Figure 5** shows his matching process.

The speaker recognition process is similar to the matching process with two main differences. Firstly, only the selected speaker profile is loaded during word recognition process because at this point, the identity of the

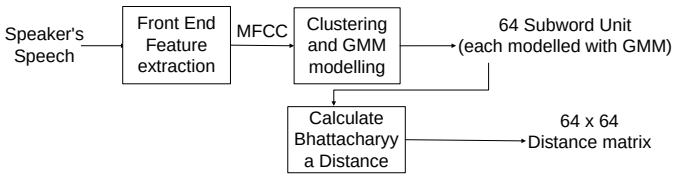


**Figure 3.** Sub-word unit generation.

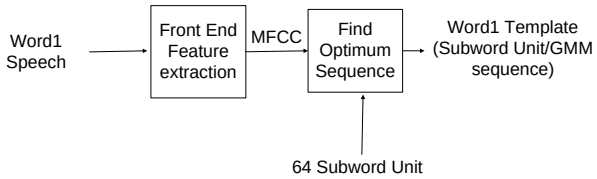


**Figure 4.** Word template generation.

speaker is already known. In speaker recognition, the speaker profile is polled and the input is compared against the respective activation keyword registration for each speaker. Secondly, instead of using edit distance, the speaker recognition process uses the posterior probability of the input given the sequence of GMM distribution in the template. This method gives us more flexibility in setting the threshold of acceptance.

## 6. Algorithm Evaluation

In this section we present the results and an objective evaluation of the proposed ASR system. We define

$$SNR = 10\log\frac{\sum_{k=0}^{n-1} S^2(k)}{\sum_{k=0}^{n-1} N^2(k)} \tag{25}$$

where $S(k)$ is the speech signal energy and $N(k)$ is the noise energy. In this ASR experiment, the noise of vehicle motor and restaurant are from NOIZEUS noise database [10].

### Speech Recognition Tests

Speech recognition test is carried out using ASR system in Section 5. The proposed blind source separation is implemented before VAD algorithm. In general, better separation results are achieved for the system with fewer speech sources than microphones [5]. In our case, we use 2 microphones to receive the mixed voices from 4 speakers. In the separated signals, one speaker's voice is selected by using automatic speaker recognition and then conducted the isolate word recognition test. The performances from SS method, ZCE method, and Entropy-Based method are compared with that from the proposed VAD noise suppression method in motor and restaurant noisy environments. Experimental results on accuracy are given in **Table 1** for the situations of SNR = 0, 5 and 10 dB. The recognition ratios under restaurant noise environment are given in parentheses.

Compared with entropy-based method that achieves the most accuracy among VAD algorithms, the relative improvement in the case of SNR = 0 dB reaches 2.5% (1.2%), while in the case of SNR = 5 dB, the rate improvement is 1.4% (0.33%). The entire ASR system works in a frame-by-frame manner and meets the real-time operation for most embedded electronic applications. In addition to the noise used in the experiment, the similar results are achieved by using street noise from NOIZEUS.

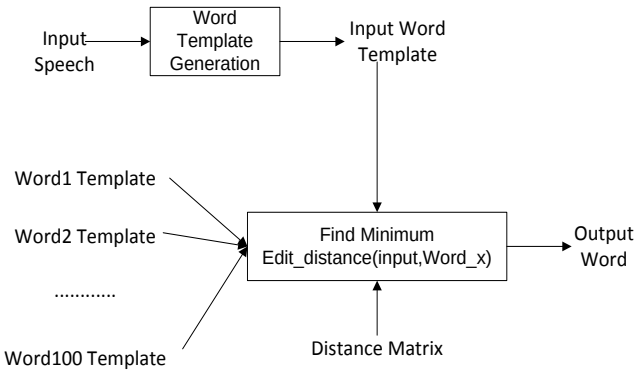


**Figure 5.** Proposed method for word recognition.

**Table 1.** Accuracy in vehicle motor and restaurant noise.

| SNR | 0 dB | 5 dB | 10 dB |
|---|---|---|---|
| SS | 60.43(58.33) | 79.86(75.94) | 92.23(88.01) |
| ZCE | 76.77(70.52) | 85.42(79.41) | 93.91(87.68) |
| Entropy-Based | 84.09(83.56) | 87.18(85.09) | 93.95(91.82) |
| Proposed VAD | 86.59(84.72) | 88.53(85.42) | 93.98(92.08) |

## 7. Conclusion

In this paper, a complete speech recovery algorithm is proposed and implemented for ubiquitous speech environment. It can effectively recover the voice of individual speaker from mixed voice of multiple speakers in noisy environment. The key feature of the proposed algorithm is that the prior information on the number of sources and estimation of clean speech variance is not needed. The threshold used to suppress noise is generated from the speech itself, which leads to the desirable ability of adapting to changing environments. Moreover, the proposed source separation and noise suppression method does not need any additional training process, which effectively reduces the computational burden. Finally, the proposed system can be easily realized in ubiquitous environment.

## Acknowledgements

The authors would like to thank STMicroelectronics Asia Pacific Pte Ltd for providing speech dataset and experiment environment.

## References

[1] Boll, S. (197) Suppression of Acoustic Noise In Speech Using Spectral Subtraction. *IEEE Transactions on Acoustics Speech and Signal Processing*, **27**, 113-120. http://dx.doi.org/10.1109/TASSP.1979.1163209

[2] Junqua, J.C., Mak, B. and Reaves, B. (1994) A Robust Algorithm forward Boundary Detection in the Presence of Noise. *IEEE Transactions on Speech and Audio Processing*, **2**, 406-421. http://dx.doi.org/10.1109/89.294354

[3] Beritelli, F., Casale, S., Ruggeri, G., *et al*. (2002) Performances Evaluation and Comparison of G.729/AMR/Fuzzy Voice Activity Detectors. *IEEE Signal Processing Letters*, **9**, 85-88. http://dx.doi.org/10.1109/97.995824

[4] Abdallah, I., Montresor, S. and Baudry, M. (1997) Robust Speech/Non-Speech Detection in Adverse Conditions Using an Entropy Based Estimator. *International Conference on Digital Signal Processing*, Santorini, 757-760.

[5] Zhang, H., Bi, G., Razul, S.G. and See, C.-M. (2013) Estimation of Underdetermined Mixing Matrix with Unknown Number of Overlapped Sources in Short-Time Fourier Transform Domain. *IEEE ICASSP*, 6486-6490.

[6] Comaniciu, D. and Meer, P. (2002) Mean Shift: A Robust Approach toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 603-619. http://dx.doi.org/10.1109/34.1000236

[7] Aissa-El-Bey, A., Linh-Trung, N., Abed-Meraim, K. and Grenier, Y. (2007) Underdetermined Blind Separation of Nondisjoint Sources in the Time-Frequency Domain. *IEEE Transactions on Signal Processing*, **55**, 897-907. http://dx.doi.org/10.1109/TSP.2006.888877

[8] Griffin, D. and Lim, J.S. (1984) Signal Estimation from Modified Short-Time Fourier Transform. *IEEE Transactions on Acoustics Speech and Signal Processing*, **32**, 236-243. http://dx.doi.org/10.1109/TASSP.1984.1164317

[9] Chang, H.Y., Lee, A.K. and Li, H.Z. (2009) An GMM Supervector Kernel with Bhattacharyya Distance for SVM Based Speaker Recognition. *IEEE ICASSP*, 4221-4224.

[10] Hu, Y. and Loizou, P. (2006) Subjective Comparison of Speech Enhancement Algorithms. *IEEE ICASSP*, **1**, 153-156.